Blender learning made easy

blender art

MAGAZINE

Shadow Showdown

COVERART Living Room - by Victor Phellipe

CONTENTS

Sandra Gilbert
Managing Editor

"Once more unto the breach, dear friends, once more.." (Shakespeare Henry V Act III) "

This quote, while a bit odd in a 3d magazine, is an apt description of my mental state every time I get to the lighting stage of a project.

To say that lighting is not one of my favorite tasks is something of an understatement. Not only has it been the most unpleasant part of any project, for many years it actually resembled a full out battle. Often taking far more time than the entire rest of the project. And more often than not, I simply gave up at a point where it looked barely passable out of sheer frustration.

Years of research, study, books, tutorials and very stubborn persistence have allowed me to finally achieve better results. At this point I have progressed from "barely passable" to "almost got it". And yet I still dread the lighting phase of any project, even more than rigging, and we all know my issues with that.

Knowing my dislike of lighting, you would think that lighting would be one of the last things I would want to discuss. Yet oddly enough, I have been looking forward to this issue. I relish adding new tips and techniques to my lighting battle "war chest". Knowledge is a powerful weapon and I will gladly take all the help I can get.

That being said, we have gathered up a very nice arsenal of tips and lighting tutorials to better arm you for your own lighting battles. So cozy on up for what is sure to be an enlightening experience ◼

sandra@blenderart.org

So, I proceeded to set up my lighting, when I noticed a new option "Show Cone".

Having decided that the best way to get up to speed with Blender 2.5, was to just start using it on projects. So for the last couple of projects that is exactly what I have done. With a bit of poking around, I got past the "stumbling around looking for familiar tools and options" stage. Which wasn't nearly as painful as I anticipated it would be.

So having figured out modeling and textures, I started poking around in the lighting options, while working on my latest project.

I was relieved to see, that for the most part, lighting still looked familiar. Which was a huge relief as I have gone to a great deal of effort to learn how to use Blender lighting options, and I was dreading the task of relearning at all.

So I proceeded to set up my lighting, when I noticed a new option under Spot Lamps. "Show Cone"

hmm.... that wasn't there before. Wonder what it is for. Yes I promptly clicked it to see what it did.

That "Show Cone" option was "highlighting" my spot lamps area of influence. While it didn't tell me how

bright my lamps were, it was showing me exactly what was in the path of the spot Lamp and what wasn't. (fig 37) Something the dashed lines kind of did, but not as visually obvious as the "Show Cone". Now I find that very cool and useful.

So far it only seems to be an option on Spot Lamps, trust me I looked once I figured it out. But then again the Spot Lamp seems to have more options than the other lamps anyhow due to the nature of their functions.

And there you go, a very pleasant surprise for me, that will make lighting a bit easier to set up and adjust ■
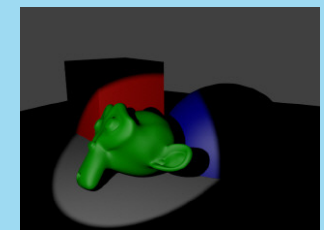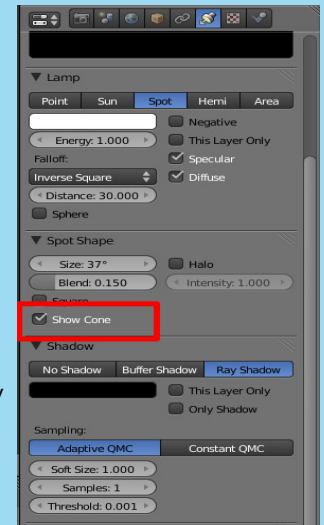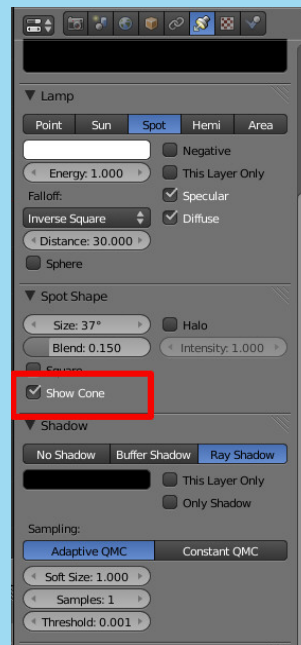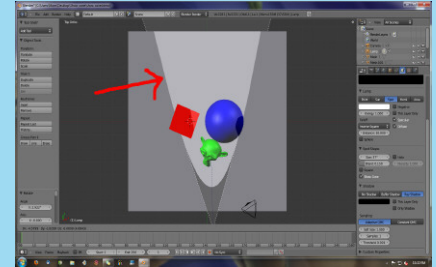
Fun with shadows

By – dreamsgate

Introduction

I have always loved shadows. They add detail and depth to scenes. Depending on how they form, they can create mysterious patterns and often create images themselves. They can also be used to great effect as a special effect to highlight or focus your attention on an object or character.

One way to use shadows as a special effect is to change their color. I am going to show you a few ways to change the color of shadows. First you need a scene of course and a lighting set up to play with. I started with a very basic light set up. There are three spot lights:

One spotlight (main spot) set up behind the camera

2 linked spots set to the sides

And a little AO for interest While this set up works rather
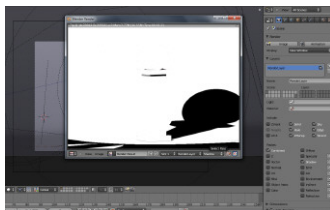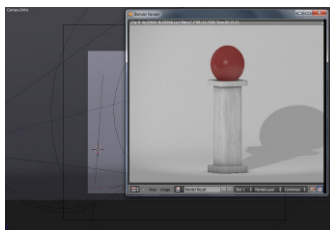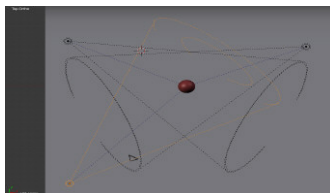
well on it's own, it is also a great starting point for a little shadow fun.

Using Nodes

I set up my render passes to give me a "Combined and a Shadow pass". Then I opened up the Node Editor and started getting it all set up.

Quick Rundown:

- Render Layer panel

- Shows the combined (original) image connections:

- Connect Image to 1st Image socket on Mix Node

- Connect Shadow to Image socket on RGB Curve Node

- 1a, 1b & 2a are Viewer Nodes to show you what each step produces.

- RGB Curve Node the RGB curve is used to adjust the color of the Shadow

- Connect Image to 2nd Image socket of Mix Node.

Mix Node

Takes the original image and the color corrected shadow image and combines them into a new image.

Okay, time to Render, making sure that "Compositing'' is checked in the 'Post Processing' panel of the Render buttons.
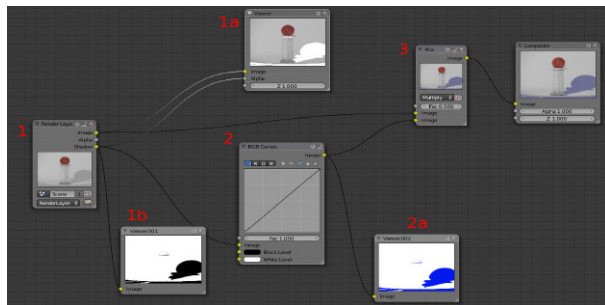
Well the shadows are definitely Blue, but they no longer have that soft blended look. Something that rendering out and adjusting additional passes (such as AO, Environment and Indirect Lighting) would take care of, in addition to perhaps a blur node or two for blending.

The fun thing about the above method, is that once you have separated out the shadow pass, you can replace it with a completely different shadow pass to create some cool special effects. Imagine an everyday character with a monster shaped shadow. How cool would that be.

Lamp by Lamp

A simpler way to change shadow color is to change it on each lamp in your scene. This resulted in a very nice effect. The shadows cast by the three spots rendered as blue, leaving the AO shadows a nice soft gray.
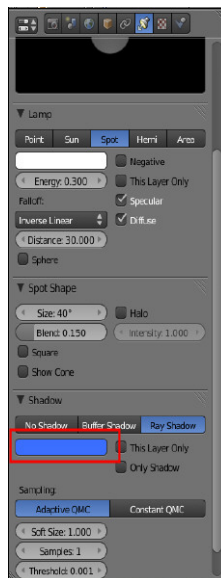
This would be an effective way to bring focus to one or two elements in your image.

Sky & Texture

But what if you wanted all the shadows to be blue. Well that is very easy too.

Change your Environment setting from "white" to "sky texture". Change your main spot light to Sun.

In the "Sky & Texture" panel select
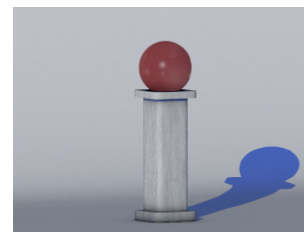
"Mountain" leaving everything else at default and render.

Now all shadows are various shades of Blue and you still have your soft blended shadows. This is just a few ways that shadows can be altered, all of the above methods can be combined and adjusted as needed for what ever you could need.

To ends my shadow fun adventure. I hope enjoyed yourself and learned a thing or two ■

By – dreamsgate

I Finally Get it! - AO

By – dreamsgate

Introduction

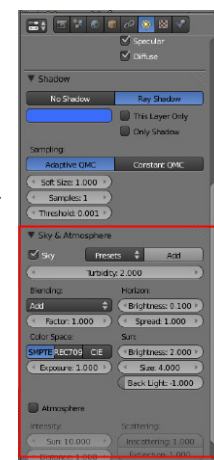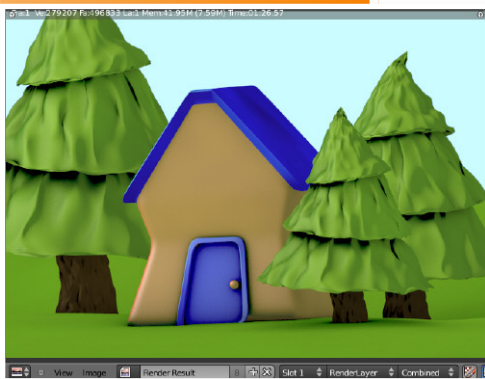Back in the day when I started using Blender, global illumination required some creative work-arounds. The most popular being a low power, duplicated spot lights parented to a 1/2 an ico-sphere.

For the time, it was a great solution. You could create nice lighting and soft over-lapping shadows with a very reasonable render time. And best of all, it was some-thing I could set up easily and use effec-tively. As time went on, coders added a new lighting option, Ambient Occlusion (AO). It created nice even shadows and added overall lighting, but it was rather slow and a little confusing to set up without a cheat sheet. So many users (myself included) stuck with the duplicated spot light method.

More time passed, and the ambient occlusion op-tions were refined and sped up considerably. Which was great, unfortunately for me, it was still rather confusing to set up unless I had a cheat sheet. And even with a cheat sheet, I generally didn't get results I liked.

So when 2.5 came out I decided to play with the lat-est incarnation of Ambient Occlusion. Hoping that somehow it might actually make sense. Oddly enough, although it still has the same options, the new lay out and grouping makes it seem far less confusing for me. Something I am attributing to the fact that now I can clearly see just what goes with what.

What was once stuffed all into one panel, now occu-pies four separate panels. So let's take a look at what we now have. This is a simple scene with no lamps added yet. We are first going to look at each section in separately.

Ambient Occlusion:

Multiply of course is going to multiply any shadows you already have. Since there are no lamps in this scene, you get a black image.
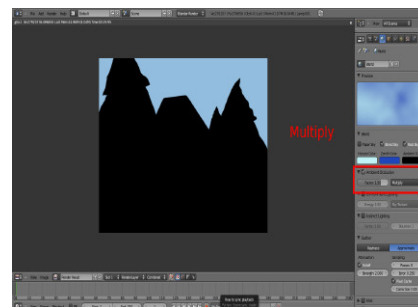
Add brings more light into your scene, so even with no lamps present you get a lighted image.
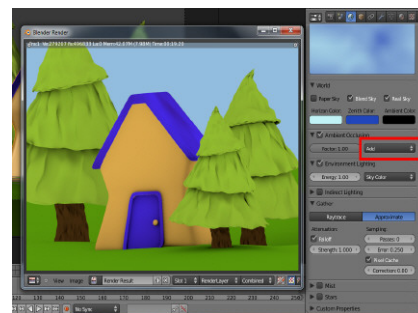
Environment lighting

Environment lighting adds light based on one of three options, White, Sky Color or Sky Texture. You can control how much light is added with the energy slider.

White adds even white light

Sky Color adds whatever color you sky is

Sky Texture adds any textures you might have set for the world in addition to the sky color, such as clouds.

Indirect lighting

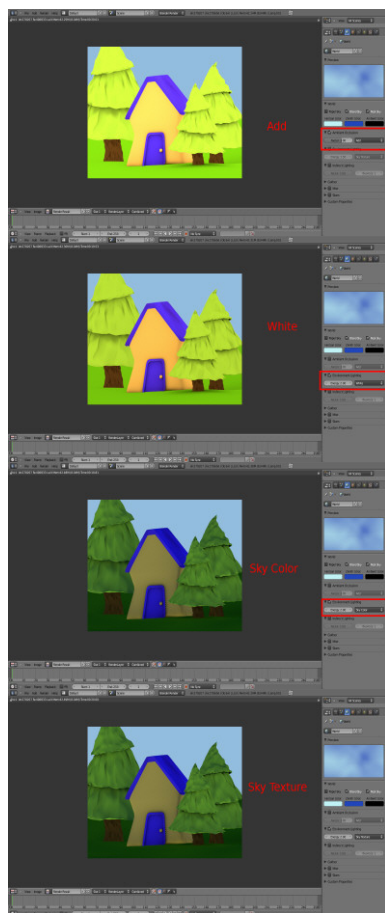Indirect lighting will provide the bouncing of light from one object to another. It does however require that light actually be present. And in our case, no lamps means no bouncing of light.
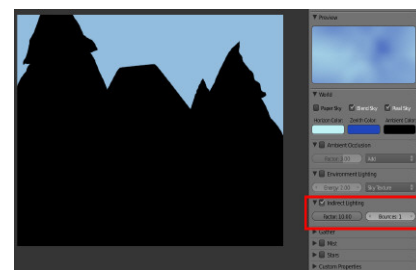
Gather

The "Gather" panel gives you two options for generating AO, Raytrace and Approximate, with Approximate of course being much faster. So it is a matter of what you prefer as to which you use for the gathering method.
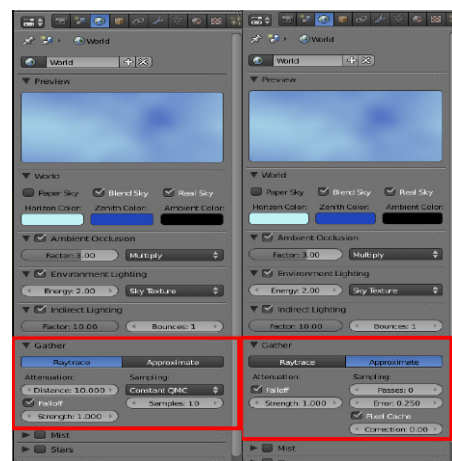
Having it all split out like this, made it much easier for me to see what effect setting changes made on my image.

Now that it finally makes a little more sense and is much easier to see what is doing what, let's put it all together.

Even with no lamps added, you already have the start of some pretty good lighting.

Of course adding lamps will add impact and allow you to high-light or focus attention on certain details of your image.

One more added bit of fun, Environment Lighting can be affected by Sky & Atmosphere settings under the Sun lamp settings. So go ahead and add a Sun Lamp, position it anywhere you want.

Scroll down to the Sky & Atmosphere panel and toggle it On, there are a couple of presets that you can chose from, or you can experiment and come up with your own.

In the Environment Lighting panel, set it to either Sky Color or Sky Texture and render.

Since there is now a lamp in the scene, you may want to lower your settings in the AO, Environment Lighting and Indirect Lighting panels, especially if it seems to bright.

Well there you go, doesn't that seem so much easier? ■

By – dreamsgate

Mesh Lights

By – dreamsgate
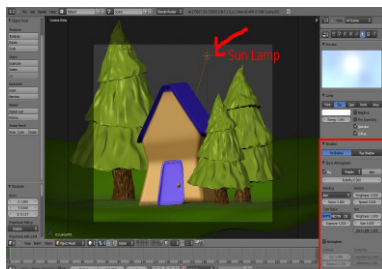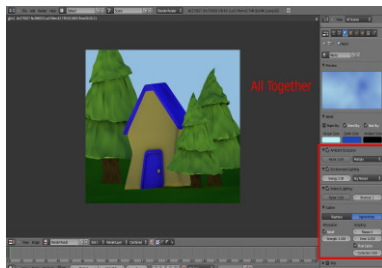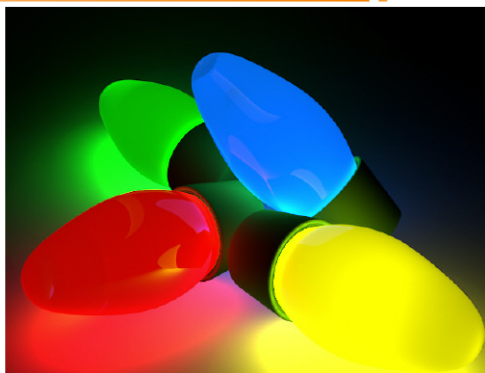
Introduction

As an artist, what could be more fun than making objects glow and cast light without having to worry about where to skilfully place appropriate lamps. In my opinion, not much.

The addition of Mesh Lights allows an artist to make any object they have created glow with a mysterious magical light.

I foresee numerous fairy lights, swords, jewels and magical items glowing their way into a fair share of future images.

And while magical items that glow are too fun to ignore, mesh lights can also provide that added touch of realism to everyday objects. Something we will all get around to exploring once we have gotten over the "oooh, it glows" phase of playing. I figure most of you should be ready for that by next Christmas (maybe :P). So let's make some Christmas tree lights.

(Jonathan has an excellent tutorial for modeling realistic Christmas tree lights at BlenderCookie.com)

We are going to start off with a simple model of a Christmas light.

Next up we add some materials. I created a simple matte green for the light base and a fairly simple red glass for the bulb.

The red glass is only slightly transparent, as most of the
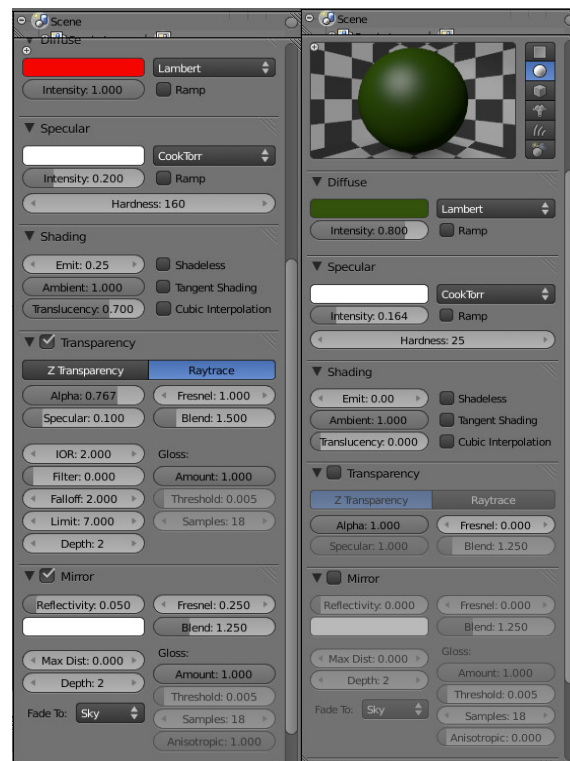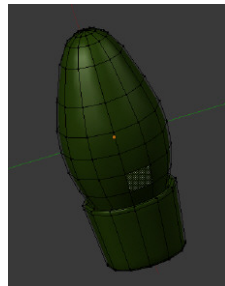
Christmas bulbs I have seen aren't overly see through. The most important part of this material is actually the "Emit" value. That is what is going to help it glow. I set mine to 0.25. I did a lot of testing and decided I preferred lower values. But I encourage you experiment and find a setting that appeals to you.

I have the Christmas light sitting on a simple backdrop, so we can see the light being cast by the Christmas light. The backdrop has a default grey material that I lowered the Spec values on.

Okay our Christmas light is ready, just one more setting to create our glowing masterpiece.

- Over in the World buttons, scroll down to Indirect Lighting and toggle it on.

- I set the Factor to 50.00. Again that was a personal preference, you might want more or less than that. And Bounces to 2.

Right under the Indirect Lighting panel is the Gather panel. The settings I used are:

Approximate Falloff of 2.00 Pixel cache toggled on

Okay time to see what we ended up with.

Well look at that, it glows and casts a nice light. But just the one light and the fact that the light is red, makes the image look very dark. A little added light would improve this image.
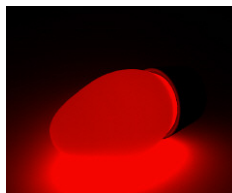
So let's add a little AO and Environment Lighting.
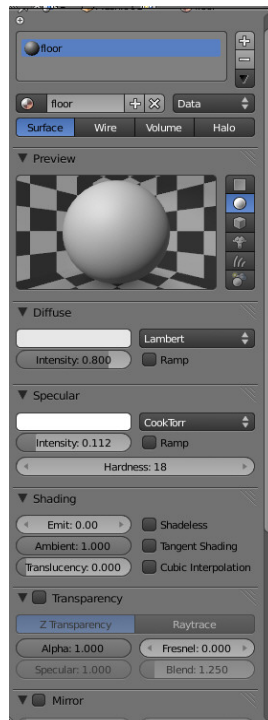
You'll notice that I set the Environment Lighting rather low. I just want to lighten things up while still being able to see my glowing red light. (fig light with AO). Well that looks a little better.

But I think that one Christmas light sitting there all by itself, looks rather sad and lonely. So duplicate several more. You can change the colors if you want for more visual interest.

Since the added Christmas lights will of course add more light to the scene, go ahead and turn AO. and Environment Lighting Off.

When you have the lights arranged in a nice way, go ahead and hit that render button.

Well that is pretty enough to use as a Christmas card ■

By - dreamsgate
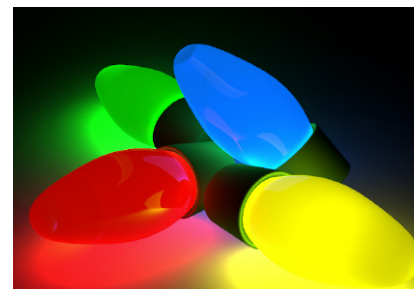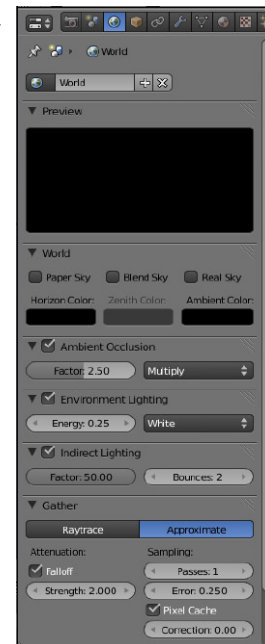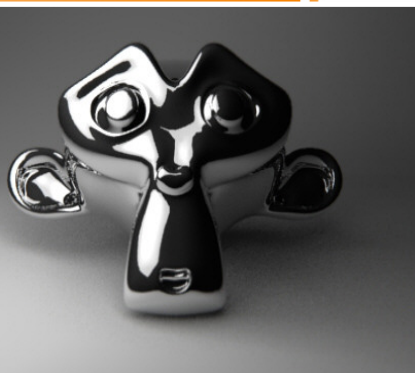
Studio Pack Shot

By - Francois Deretz (DEF13)

Introduction

I'd rather spend my time reviewing settings for studio lighting to make a nice image of a single object, especially when you have to manage glare on chrome objects! Recognizing this loss of time, I threw myself into the design of this studio pack shot. A pack shot is a high quality photo of a product on a solid background in most cases, used to present said product in a product catalog.

This is a real illustration for a pack shot (+ soft box studio settings). This is exactly what I wanted to create in Blender ...

Pack Shot Studio for Blender is a virtual solution that simulates the effects of shooting an object: light scattering, reflections of the flash on the reflective material, adjustment of the focal blur ...

Although the process of image production has been greatly simplified, it is necessary to have some knowledge in 3D and the Blender environment to better understand Pack Shot Studio. Knowledge about studio photography is a plus.

File content

You will find in this file:

3 lamp types with each one having at least one Target to aid targeting, adjustment for distance and energy and adjust color of the light in the 3D view.
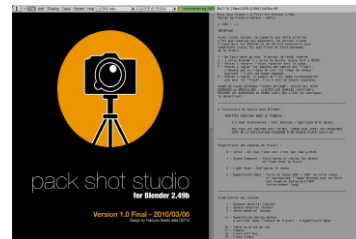
- A "Flash" like a "Soft Box" which includes an adjustable light box (square or round).

It's good for soft illumination without specular effect and maximum diffusion shade.

It also creates a square or a round reflection on object.

- A "Flash" studio with adjustment outreach directly in the 3D view.
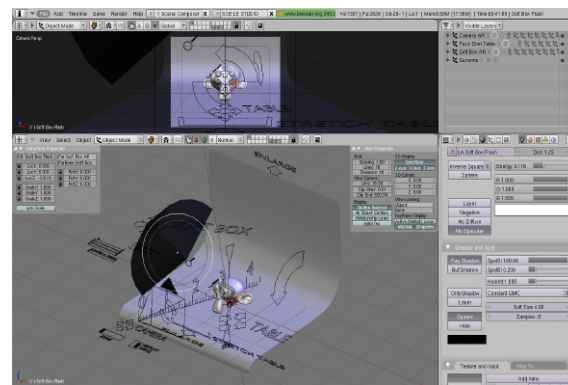
It's a good way for lighting effects mixed with specular shading and rough shading.

By – François Deretz (DEF13)

- A neon "Flash" studio

For soft lighting and specular both, perfect for a garage effect on a car body.

It creates a neon tube reflection and a relatively strong linear specular reflection on object .A customizable shooting table.

- A Camera with a target, an Auto-focus (coupled to Target or manual) and a zoom ring.

- A node for development (rendering) that allows the management of focal blur based on the settings of the camera. It also adds a slight defect to the lens to accentuate realistic rendering.

File Architecture

First, let's see how I structured my file.

We have to take a unit for 3D / perspective = 1 unit of local measurement.

We will use metric system, so a unit is 1 meter.

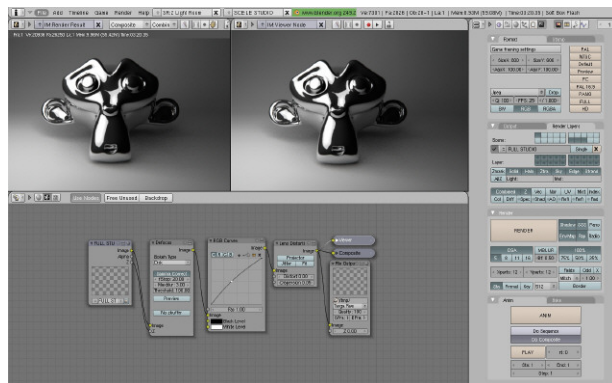In a case where the scale of your object is different from the studio, scale the import object in the studio.

Set the shadow samples of "flash", do not abuse this setting (make it too high) because of rendering time! A sample from 4 to 8 is a good average.

Consider the sample settings of the Ambient Occlusion, especially recommendations for the "flash". Values from 4 to 8 are good averages.

Let's start by opening the file. First, check that the settings are all enabled for replication ([i] User Preferences› Methods Edit› Duplicate With Object) to avoid problems when duplicating a standard "flash" (Shift + D).

The Studio consists of three areas of work and information on where we are. To select a workspace simply activate the menu with the prefix [SR:].
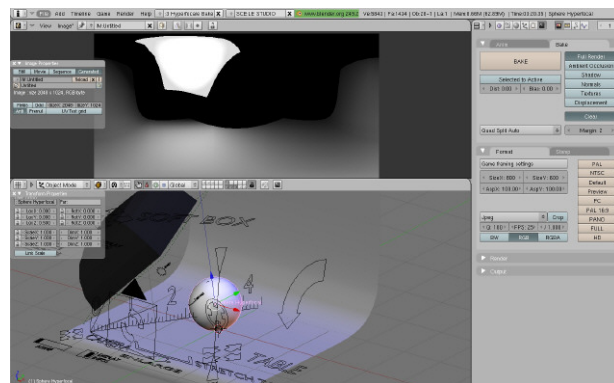
1. Composing the Scene

Let's enter the scene, choosing the scene menu and select scene 1. This space is used for positioning objects and settings and flash in the studio.

By – François Deretz (DEF13)

Activate the layers 1, 16 and 18, show suzanne, table and flash soft box.

What makes up the layers in the Studio?

1. Suzanne Material Tester

2. Sphere Material Tester

3. Glass Material Tester

11. Hyperfocal Baking Sphere. For use in the workspace: 3 hyperfocal Bake

16. Shooting Table: Visible Objects frame = Table / Object Table

17. Camera: Visible Objects = Frame Camera / Camera CAM / Z Subject meter

18. Flash Soft Box: Visible Objects frame = Soft Box / Object Soft Box / Lamp Soft Box

19. Direct Flash: Visible Objects = Flash Frame Direct / Direct Flash Object / Lamp Direct Flash

20. Neon Flash: Visible Objects =
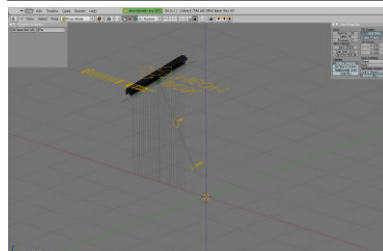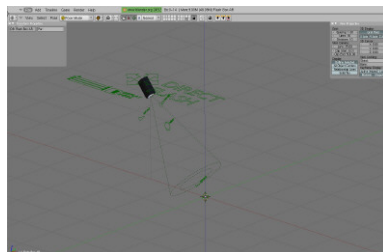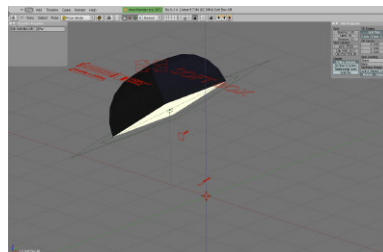
Frame Neon Flash / Flash Object + Map Dupli Spec Neon / Neon Flash Lamp (no spec) / Neon Flash Lamp Spec (no diffuse)

How to duplicate a group flash?

Method 1: Simple Duplication

The flash will duplicate independent settings of the duplicated flash.

- Select any object that makes up the "flash" to duplicate.

- Make Shift + G› In Same Group or Object In Same Group.

- Make SHIFT + D: All the information from the "flash" is duplicated.

Note: I strongly recommend starting the selection process from the frame.

Method 2: Multiple Copying

A single "flash" (parent) controls a set of "flash" (children) of the same type.

- Create any object on any layer, an Empty is the simplest.

- In the panel: Window Buttons› Object (F7)› Anim Settings and activate DupliGroup.

- In [GR:], type GR "The Name Of The Flash. example: GR: GR Soft Box

- The DupliGroup appears behind the object that duplicates it.

By – François Deretz (DEF13)

- Create a plane that you subdivide (be reasonable in the subdivision, avoid large values).

- Parents who used to be DupliGroup this plane.

- Keep only the plane selection.

- In the panel: Windows Buttons› Object (F7)› Anim Settings and activate or dupliverts DupliFaces.

- Your flash original is duplicated for as many vertices or faces that make up the plane. The settings of these clones will be totally dependent on "flash" original.
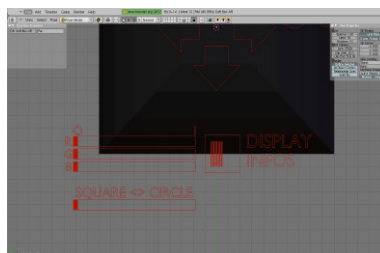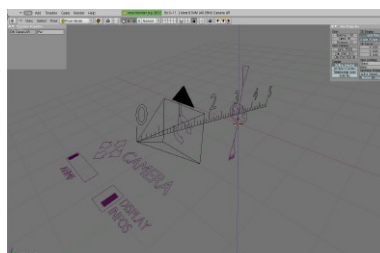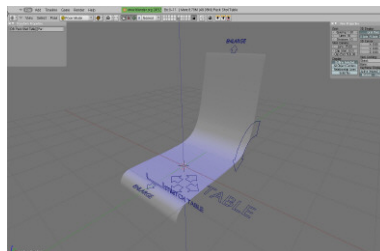
2 Light Room : Configuration space of the initial rendering and rendering composite.

3 Hyperfocal Bake : Make a record 360 ° x 360 ° of your scene and save the resulting image into a background image HDRI.

This process will be a little long depending on the size of the rendering.

Conclusion

I hope you will enjoy using this studio. I still have to adapt it to Yafaray and Luxrender and make a version for Blender 2.5x (too many bugs for the moment - even with the Alpha 2). You can follow progress on the topic of BlenderClan ■

François Deretz (aka DEF13)

I was born in 1976 and I started my career as a graphic designer in 1998. I am now a freelance art director and with the help of my wife, am building a little visual communications agency specializing in the fields of watches and jewelry -(http://www.cb-visualdesign.fr ; http://www.flickr.com/photos/def13_marseille/sets/).

I discovered Blender version 2.26 and included it in my working process during the release of Version 2.32.

We pack a lot of shots in our studio and took reflexes management of light and reflections that I could not find in Blender. That's how the idea was born of a file "Photo Studio" in Blender
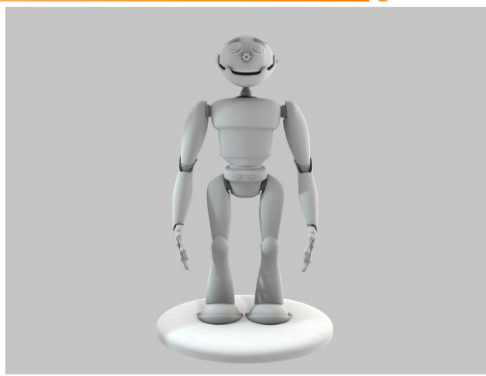
Robbie the Robot

By – Daniel Hand

Introduction

Modeling of mechanical objects in blender is a great way to improve your modeling and rigging skills and is often easier than modeling purely organic objects. In this tutorial, I will guide you through a high-level approach to creating an anthropomorphic robot character called Robbie.

Reference Images and Sketches

I used several sources of inspiration when sketching Robbie, ranging from movies such as "Robots", by Twentieth Century Fox to the almost limitless supply of images provided by Google. While the reference movies and images helped develop the sketches, I found small model robots, similar to what you may find in a toy shop, the most useful for understanding how the various joints worked.

Figure shows the final sketch that I used as the basis for the model. While the sketch is quite basic and is posed differently to the final render, the essential characteristics and relative proportions should be clear to see.
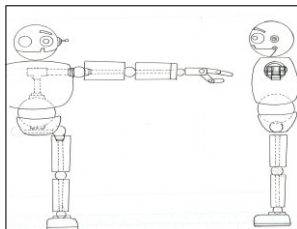

One of the decisions that I made early on in the design process was that Robbie would be a friendly looking robot and have certain human-like qualities.

It is for this reason that he has large eyes and his jaw was modeled to appear like he is smiling. Anatomically, he loosely matches that of a human, but you'll notice that his arms are disproportionate in size with his legs and body.

The reason for doing this was two fold, firstly I wanted to give him a bit more character and experiment with the idea of him being a robot. The second reason was to make future animation of the model easier as it's far more difficult to convincingly animate a model with exact human proportions.

Modeling the Skeleton Framework

The modeling process started with creating the basic skeleton framework. While most of the skeleton would later be hidden from view, I used it as a guide when modeling the visible parts of Robbie and later in the rigging process.

Each component of the skeleton frame started off as a simple primitive object such as a UV-Sphere or cylinder. Placement of the ball and socket joints was achieved using Blender's "Snap Cursor to Object" and "Snap Object to Cursor" operations.

The ball and socket joints were created from a UV-Sphere with 12 segments and 12 rings [12,12].

By – Daniel Hand
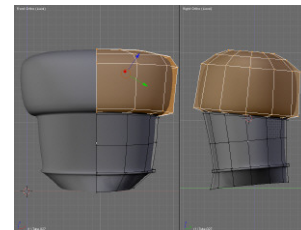
The socket joint was modeled by selecting the third edge-loop [Alt-Right Mouse Button] as shown in figure 002, duplicating [Shift-D] it, separating [P] it from the parent object and then using a series of extrudes [E] and scaling operating [S, Shift+X] to create the tapered socket joint as shown in figure 003. Creases were added by placing loop-cuts [Ctrl-R] in close proximity to existing geometry. Smoothing was added to both the joints and sockets and a sub-surface modifier was added to the socket joint.

After creating the first socket joint, it was then duplicated [Shift-D] and rotated [R] about the centre of the ball. Each ball and socket joint in the model is simply a scaled linked duplicate of these th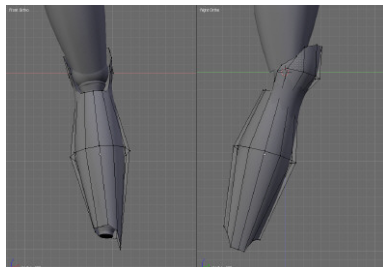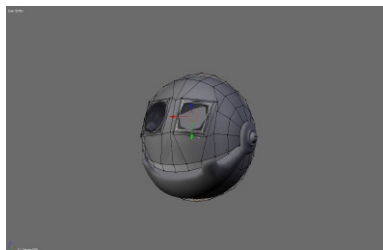ree objects. Using this simple ball and socket joint, together with open cylinders consisting of 12 segments, I created the skeleton framework as shown in figure 004.

Modeling the Head

I wanted Robbie to possess human-like characteristics and to look friendly. I therefore decided to give human-like eyes and modelled his jaw such that it appeared that he was smiling. The head started life as a UV-Sphere[12,12]. Half of the vertices were removed and a mirror-modifier about the x-axis was added.

I created the jaw by selecting the faces shown in figure 005, duplicated [Shift-D] and separated [P] them and extruded [E] the region. Additional detail was added to the jaw by adding loop-cuts [Ctrl-R] around the axis of rotation as shown figure 006 and extruding [E] a face to resemble a tooth.

I'm not not going to cover how to create Robbie's eyes in this tutorial as there are some great eye tutorials available on-line and in previous publications of Blender Art magazine.

I did however enhance the eyes by adding a bottom eyelid and placed a tapered cylinder above each to act as an eyebrow. When creating the eye socket, the aim is to create a realistic hole in the mesh without generating ugly geometry. I was able to keep the mesh clean by deleting [X] one of the faces and selecting the resultant edge-loop [Alt-Right Mouse Button], snapping the cursor to object [Shift-S] and selecting "To Sphere" from the "Mesh Tools" menu. The edge-loop was then scaled slightly to accommodate the eye as shown in figure.

By – Daniel Hand
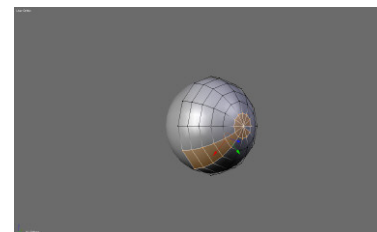
Robbie's nose was created using the bolt factory wizard located in the scripts menu. The default settings create a bolt with a lot of geometry, especially around the shaft of the bolt. As most of this will be hidden from view, I deleted it and replaced it by extruding [E] the edge-loop where the shaft meets the head of the bolt.
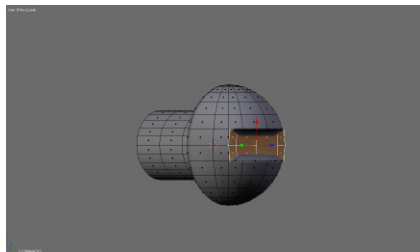
The head was completed by adding a low-poly bolt to each side. The low-poly bolt was created from a UV-Sphere[12,12] where half of the sphere is deleted and the largest edge-loop extruded and scaled inwards. It was then extruded once more to form the cylindrical section of the bolt as shown in figure 008. The slit in the top of the bolt was created by extruding [E] some of the faces inwards along their normals.

Modeling the Upper Body

The upper body started life as a cylinder[12] with its major axis aligned along the z-axis. Once again, half of the vertices were removed and a mirror-modifier about the x-axis was added. A succession of loop-cuts [Ctrl-R] added along the length of the cylinder along with scaling [S,Shift-Z] formed the basis for the object. Once I was happy with the proportions along the z-axis and x-axis I switched to the side-view and scaled all the edge-loops except the ones at the top and bottom of the cylinder.

This improved the proportions of the upper body. Creases were added as described earlier in this tutorial. Once I was happy with the general geometry of the up-per body I selected the upper section as shown in figure and separated it from the lower section. This would later allow me to place Robbie in a more natural pose by placing an arch in his back.

A plate was added around Robbie's neck to provide a little more detail to the model. This was comple-mented with a few low-poly bolts as used earlier on Robbie's head. A close up of the plate is shown in fig-ure.
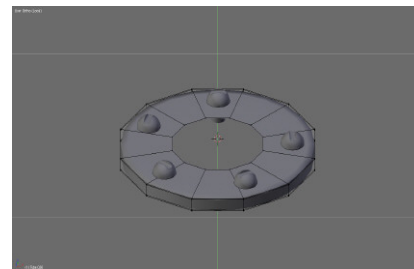
Modeling the Lower Body

Robbie's lower body was created from a cube with one level of sub-division [W] and all vertices with negative x or y coordinates deleted and a mirror-modifier added for both the x-axis and y-axis.

The cube was positioned as shown in figure and the base of the cube scaled [S,Shift-Z] inwards while being locked in the z-axis. A total of seven loop cuts were added to the cube to create the ge-ometry shown in figure 011. The lower body was completed by add-ing four low-poly bolts, two to the front of the object and two to the rear.
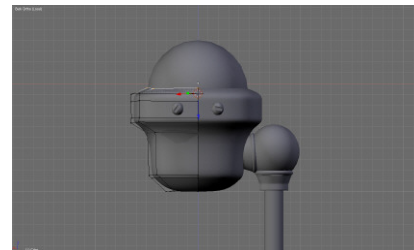
If you look carefully at the bolts, you'll see that each has varying degrees of rotation to increase realism. The bolts not only help to complete the model but also create some nice shadows during the render stage.

Modeling the Arms and Hand

The upper and lower arms were each created from a cylinder[12]. Three edge loops [Ctrl-R] were added to the cylinder and scaled [S] outwards to add some definition and a smooth contour. Additional geometry was added to the upper-arm around the shoulder by selecting the six edges as shown in figure 012 and extruding [E] them three times while tracing out a curved surface around the shoulder joint. Additional geometry was added to the lower arm in exact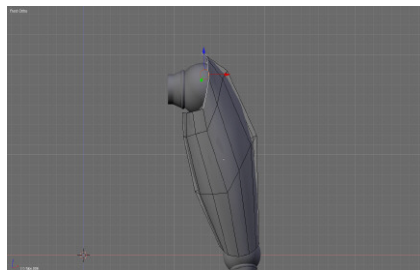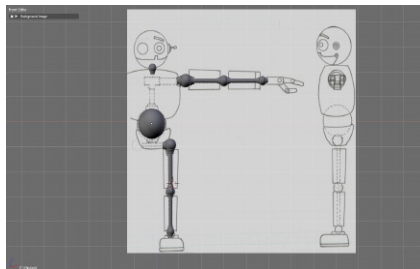ly the same way as shown in figure, but this time it was to protect the elbow joint. Minor adjustments were made to the arm while viewing the model from various angles and finally a sub-surface modifier was added to each object and smoothing applied to each of the faces.

Robbie's hands are based on a flattened Cube with two edge-loops [Ctrl-R] placed at the locations of each of his three fingers as shown in figure 014. Additional geometry was extruded [E] from the side of the cube where the thumb would later be attached. The fingers and thumbs are each based on Robbie's arm skeleton, with an additional tapered cylinder[12] added to each to form the finger tip. Each of the digits were attached to the hand and placed in a natural pose.
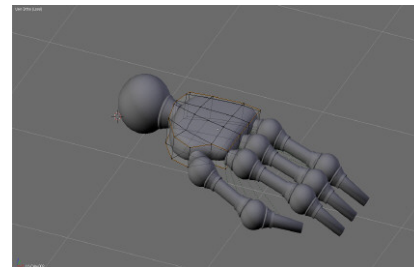
Modeling the Legs and Feet

The upper and lower parts of Robbie's legs were created in exactly the same way as Robbie's arms, actually, I duplicated [Shift-D] the arms and tweaked each of the edge-loops to get the desired geometry. The only minor difference between the arms and legs is that the legs bends in the opposite direction to the arm and therefore it was necessary to rotate the lower section of the duplicated arm. A detailed view of the geometry of the Robbie's leg is shown in figure.

By - Daniel Hand
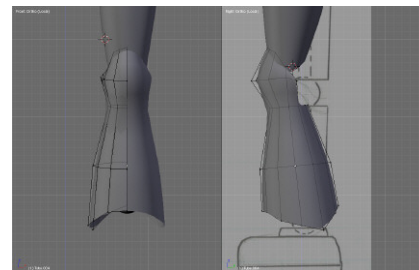
Moving on to the last part of the model, Robbie's feet. It should be of no surprise that each foot was created from a UVSphere[12,12] with half of the faces deleted [X] and additional faces added to form the sole of the foot. A sub-surface modifier was then added along with an edge-loop around the base of the foot to create a solid crease and flat sole.

The foot was then scaled [S, Y] along the y-axis to elongate the foot and give Robbie greater stability. The toe-cap on each foot was created in a similar way to how I created Robbie's jaw. The faces shown in figure 016 were selected, duplicated [Shift-D], separated [P] and extruded [E] to create the new region. Additional edge-loops [Ctrl-R] were added to the toe-cap around the axis of rotation and finally sub-surface modifiers and smoothing was added to each part of the foot. As a finishing touch, I added two low-poly bolts to each foot, one on either side of the foot to give the foot the same level of detail as Robbie's jaw.
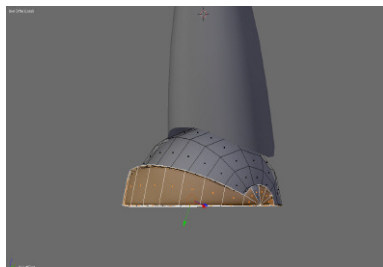
Finishing Up

The final step in the modeling process was to duplicate [Shift-D] all the objects on Robbie's left hand side and mirror them about the x-axis [Ctrl-M,X], while ensuring that the 3D cursor was located at the origin.

So there you have it, a high-level guide to creating an anthropomorphic robot character called Robbie using primitive geometrical objects in Blender.

For those of you interested in watching a series of video tutorials going through each of the stages outlined in this tutorial, as well as rigging, lighting, UV-mapping, texturing and composite rendering of Robbie, please subscribe to my Vimeo channel "Robbie The Robot" http://vimeo.com/danieljhand/channels ■

By - Daniel Hand
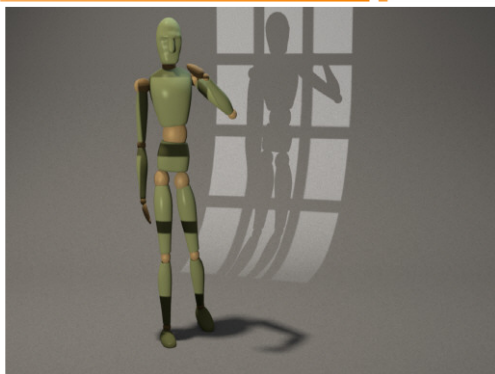
Textured Lights

By – William Meador

Introduction

Textured (or dappled) lighting provides a scene with extra depth and interest, it can enhance the mood of a scene, and it is an opportunity to expand a space by letting the audience know that there is more to it than what is just seen in the camera's view. Textured lighting in the real world can be seen as sunlight passes through clouds, trees and plants, and architectural elements, such as arbors, glass and steel walls, truss work, etc. This article outlines the real world technology used to create dappled light and provides insights into using these ideas in a blender scene using texture projections.

Real World Textured Light

Common terms for texturing light in the entertainment industry are "gobo" and "cuculoris" or "cookie." Don't let anyone tell you they know what gobo means. Some say "goes between," others say "goes before (the objective lens)," but just know that it means a pattern that is projected with a lighting instrument. Gobos specifically need an ellipsoidal reflector spotlight (ERS). An ERS has the gobo, and shutters to crop it, at the second focus of its ellipsoidal reflector. The light source is at the first focus. Add a lens that can be adjusted towards and away from the gobo and one has an instrument that can project an image as sharp or diffuse as desired. Cookies generally sit in front of a lighting instrument rather than inside the instrument. Sharp or diffuse qualities of the projection are based on how close or far the lighting instrument is from the cookie. Gobos tend to be just a few inches in diameter, whereas cookies are often as large as a few feet on a side.

Virtual World Textured Light

Not unlike the real world, in the virtual world textured lighting can usually be added in two ways. One is to pass light through geometry and let the geometry and materials cast shadows into the scene in a similar manner to the cookie. The other is to use projections via textures in a light source, which is similar to gobos. Passing light through geometry works well, but can be computationally expensive and it can often take a lot of time to create. Using projections requires some work in an image editing application, but little else is needed to get them to work. This article focuses on using projections for textured lighting.

Textures

What should a gobo/light texture look like? In theatrical lighting it is often a silhouette only, which gives a sense of some object casting shadows. See theatrical gobo suppliers, such as Rosco, Lee, Apollo, and Great American Market (GAM) for examples. It is possible, however, to use colored images as projected textures. This might be useful for simulating slide/film/video projectors.

To create a texture for projecting it is best to keep two things in mind. One is to consider what in a scene would cast shadows, such as trees or window details. The other is to consider the quality (or diffusion) of the shadow being cast.

By – William Meador

Objects that are very far away from the surface that is receiving a shadow will cast a diffuse shadow, while an object closer will cast a sharper shadow. Simply blur the texture in order to get the proper quality of shadow. Textures designed for shadows should have white or the light color pixels for where light will pass through and black where there should be shadow.

Assigning textures to spotlights is a matter of selecting the light, going to its texture properties, assigning a new image texture and making sure its mapping is set to "View."
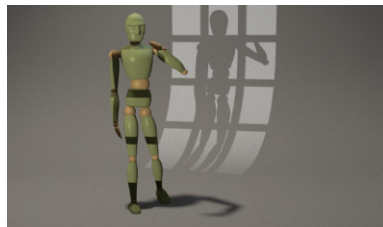
Examples

A straight tree gobo image projected onto the scene.

A blurred tree gobo to achieve the proper shadow quality.

A window gobo projected onto the scene.

Window and tree combined. Figure 4

What if the lighting is great and a textured source changes the intensity too much? Try a projection with a negative energy source, but be sure to invert your image texture. Negative projections have the bonus of not needing to cast shadows. Figure 5

Caveat! Texture can not be applied to halos. Let's hope this changes some day…
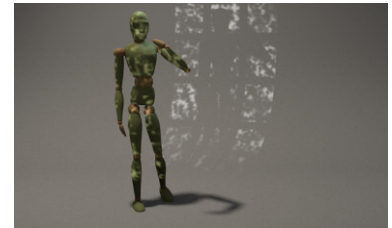
Window and tree combined. What if the lighting is great and a textured source changes the intensity too much? Try a projection with a negative energy source, but be sure to invert your image texture. Negative projections have the bonus of not needing to cast shadows. Figure 5

Caveat! Texture can not be applied to halos. Let's hope this changes some day ■

Lighting the Vault

By – Gaurav Nawani

Introduction

In real life nothing remains untouched by light, every aspect of the visual medium that we savor is due to the presence of light. Light as an element would mean nothing if its interference with colors, texture, shape volume and distance is somehow disconnected from it. The light also plays significant role in drawing the viewer's emotional attention towards the scene.

We humans have incredibly sharp understanding of light and its effects so much so that the unconscious brain lets us enjoy the beautiful picture or landscape without even thinking about the light properties that created it. That deep connected understanding is ingrained in to our minds since the day we had opened our eyes to the beautiful world of light. Any little deviation in the regular order of light such as placement, direction, intensity and or color instantly alerts our mind to pause and re-think over the scene, for example red light is considered a signal for danger, similarly deep blue for fear or the unknown/mystery and both of them are used in the movies and 3d cg scenes. We in the world of CG remain mystified and starry eyed about the visual richness of our world, being awed and inspired we turn to the tools at hand in the CG world to create that exclusive realistic feel.

In this walkthrough we are going to show you step by step how we created the lighting setup for one of the scene of our upcoming sequel to the game Pahelika. The scene in question is titled "The Vault". According to the design documentation this scene was to be shown as an old well built hidden chamber containing various vaults and a backup generator. The room has only one approach and that is through the lift.

Step1. The setup

After the modelling process was completed we moved on to lighting the scene. Now first step was to find out the light source, we had a tungsten bulb fitted in the room

that took care of the primary light. An Omni lamp is a natural replacement for the tungsten bulb in real life. Next shadows were enabled on that lamp and the scene was checked for aesthetical balance.

Step2. A problem

The first thing we need to deal was that since the bulb is placed on one wall, the wall perpendicular to it is lit more than the wall it is on, which looks odd as human eyes are well trained to catch this anomaly, so to correct this one lamp was added on a new layer also the wall which is receiving less light is moved to the same layer.

By – Gaurav Nawani
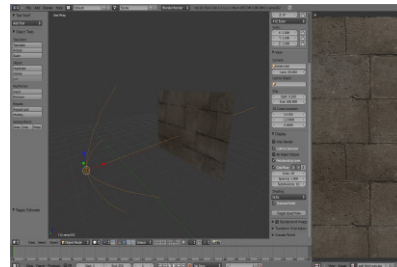
Now we changed the lamp to account for lighting in its own layer so as to protect its light interaction with other objects in the scene. Since we do not need any specularity in the scene we also went ahead and disabled specularity and let only the diffusion affect the wall.

Default intensity was also reduced to offset the brightness when both the layers were rendered in combination.

Step3. Fake Global Illumination

We are not using AO in the scene to manage faster rendering times.

We decided to add more lamps to compensate for the possible indirect lighting. So another lamp is added almost opposite to the main light source. We are going to use this as a filler lamp so we will need to reduce the intensity so as to match a near realistic physical diffusion of light.

Step4. Details

Strange as it may sound to a newbie, most scenes in 3d are in more than one way a plain copy of the basic photographic light principles. Since we already had two light source a main light and a fill light, the only one lacking is a highlighter. Now as those of you who had read about the basic three point lighting will understand the value of an highlighter lamp. It brings out the depth by increasing contrast over the edges of the objects in the

scene making it more appealing to the viewer. So we added a lamp and disabled diffusion on it and placed it diagonally opposite to the direction of light and shadow. Now as you can see it brings out decent highlights over the objects bringing them forward in the scene.

The scene is not finished and it pending for final review. But I hope that it have shown you how to proceed in lighting solution for this scene. Lighting is a very vast area and one can never learn enough to be expert however to begin learning is the first step indeed. All of you who are new to the field of CG lighting are encouraged to purchase the book 'Lighting & rendering' by Jeremy Brim which we have already reviewed in the earlier issue of blenderart magazine ■
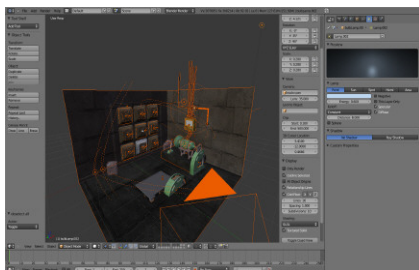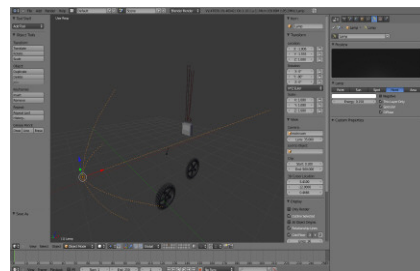
Lighting Troubleshooting

By – dreamsgate

Unless you are a master lighting guru, at some point you are going to run into lighting problems. Lighting problems are generally (but not always) one of the following:

- really bright areas
- really dark areas
- strange shadows
- lines on your objects

Here is a short check list of suggestions for troubleshooting what has gone wrong.

Deactivating Materials

- Materials obviously interact with light, and can often be a cause of confusion when trying to track down lighting problems. So to isolate if it is a lighting problem or more of a material problem, try temporarily deactivating all materials.

- Create a default material, a medium gray one, and enter the name into the Material field, Render Layers panel. Just clear this field to return to your original materials.

- After rendering check to see if you have the same lighting problems with just gray objects. If the problem has disappeared, you have a
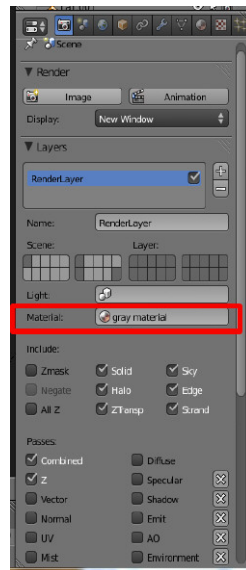
materials-interacting-with-light problem.

- Check the material settings, especially ambient, reflection and all those little buttons and sliders on the Shaders panel. Because some lights can be set to affect only certain materials, be sure to check those if only a few of your object appear to be lit wrong.

Kill off your Lights

- Since the problem could be one or more of your lights, it is time to check them all one by one.

- Move all your lights to an unused layer and start re-adding them one at a time, checking setting and how it affects the image overall.

- As you add them back in, make sure they add in nicely, reduce light energy as needed or even considered eliminating the last added one if it doesn't seem to be adding to your image.

Check Layer only Lights

- Lights can be set to affect only certain layers, so if only some of your gray objects are appearing odd, then check to see if a lamp is only affecting one or more layers.

- This can be set accidentally, or you may have forgotten that you set it earlier in your project. Misc tips

- Negative lights can add desired shadows, but they can also introduce some extra problems. So pay extra attention when using them and make sure they are not adding to your problems.

- Overly textured lights can make your scene have random weird colors.

- Overly colored lights can affect material colors as well as your lighting in general. Try to stay with a slight tinge of blue or yellow or shades of white, or your material may show blue in the Material sub-context but render green, it can take ages to track down what has happened.

- Environment settings can cause a whole host of problems, especially the Horizon, Zenith, and Ambient light settings.

While this list obviously won't cover every lighting problem, it does give you a good starting point to track down what has happened. ■

By – dreamsgate

Blenderfarm

By – Tj Armstrong

By now, I think it's fair to say that Blender 2.5alpha has shown the industry that Blender is a serious contender in the 3D world. With features that users of other applications can only dream of being integrated in the base install, features such as; Volumetrics, Water, and Cloth/Soft Bodies, to name but a few.

It is easy, however, to forget that the UI and integrated feature list aren't the only things that are getting a makeover.

With the scripting interface being revamped, A whole new world of possibilities are being opened-up. Possibilities that with little effort could become realities, even the more remote such as 'bake-farming' and fully integrated render-farms can now be realized.

'Bake-farming', in-fact, as an idea, has previously been laughed at by some, as an unrealistic dream. But with projects such as BulletPhysics, and commercial ventures such as PhysX, starting to show that simulations can be done with stream-processors, either via OpenCL or CUDA. What's stopping render-farms from doing the same?

With render-farming becoming outdated by better, more powerful, hardware and renderers such as Lux-Render working towards OpenCL Support. Render-farms could soon be forced to change tack and evolve into hybrid bake/render-farms in the near future in an effort to retain profit levels in a closing market.

With more complex simulations taking a number of hours to complete, having a simple button that submits the work to a farm and returns the baked data within minutes, would be highly beneficial. Especially when, as with many simulations, you may want to tweak the settings many times before you are truly happy with the result.

At the time of writing, only two render-farms have confirmed support for Blender 2.5 and subsequently Blender 2.6 on release.

With the "Global-recession" advancing, the power of Amazon's EC2 "Elastic Compute Cloud environment" and the constant increase in the speed of end-user hardware, it could be fair to say that render-farms should be looking to enhance the service they provide, in order to retain, or even increase profit margins. Yet none of those contacted prior to the writing of this article admit to working on enhancing the service they provide. Either by redesigning their upload method into an easy-to-use augmentation of Blender's render UI, Or by investing in the development of 'bake-farming'. While some are making promising progress on their own upload systems, they are still very much external operations that disrupt the production process in a way that is no longer necessary.

It appears, that for now, the future of a fully integrated "Blender-Farm" is still a very distant dream.

Thankfully though we can content ourselves, at least for now, with the multitude of integrated 'internal-external' render engines that are expected as new features in Blender 2.6.

'Bake-farming' is, as the name suggests, a way of baking a simulation via a render-farm-like system.

The simulation would be submitted to the farm, that would then calculate the simulation's outcome, and return the baked data in a far shorter time than the individual machine could have done. Much in the same way that a render-farm works.

The theory that makes 'bake-farming' possible is already being put to practice in stream computing environments ■

BLENDMAN 2010-03
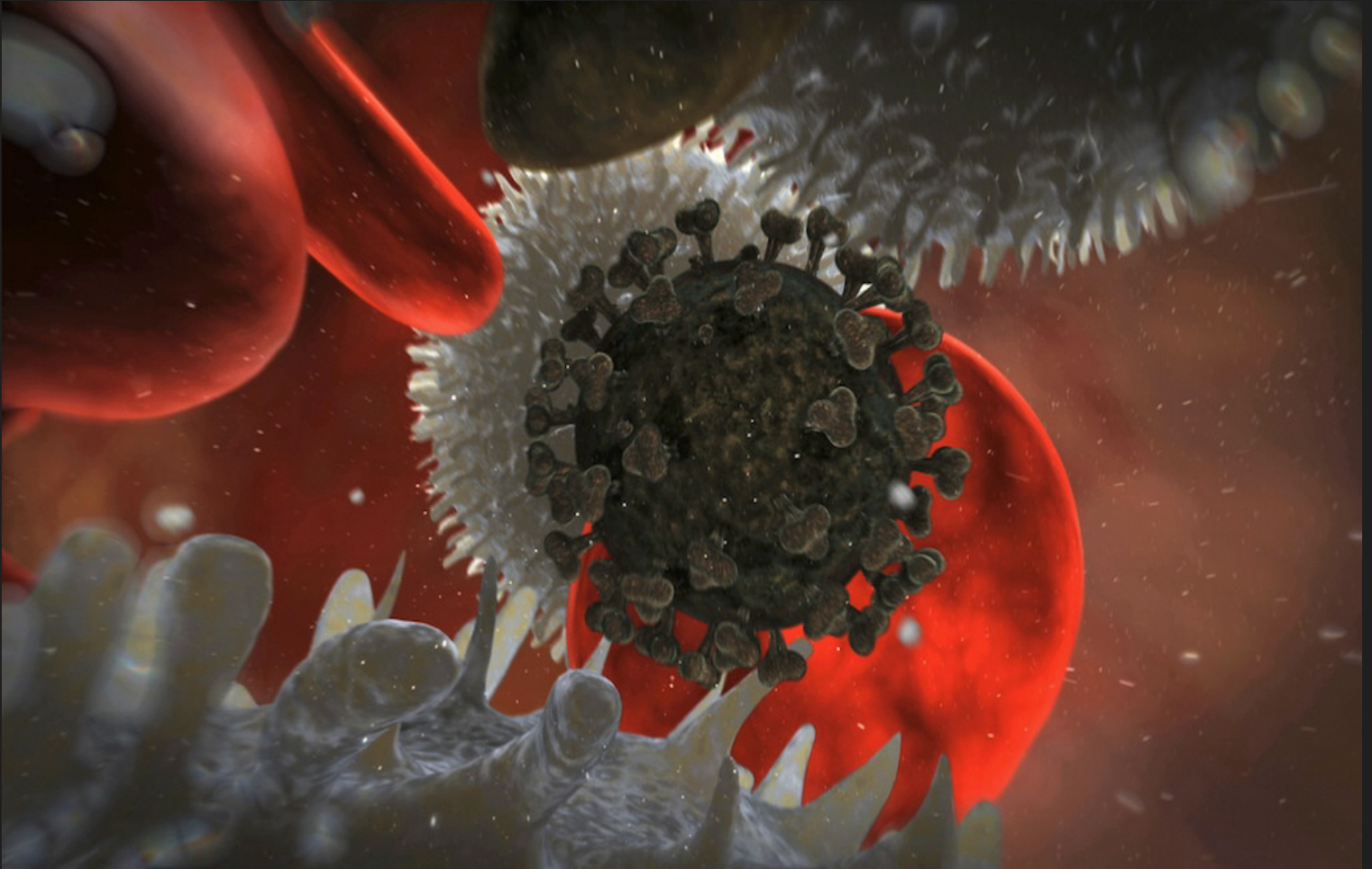
Blender 2.49
Yaf(a)Ray 0.1.1
Copyright © 2009 Victor Phellipe
http://victorphellipe.tk/

Here is how!

1. We accept the following:
- Tutorials explaining new Blender features, 3dconcepts, techniques or articles based on current theme of the magazine.
- Reports on useful Blender events throughout the world.
- Cartoons related to blender world.

2. Send submissions to sandra@blenderart.org. Send us a notification on what you want to write and we can follow up from there. (Some guidelines you must follow)
- Images are preferred in PNG but good quality JPG can also do. Images should be separate from the text document.
- Make sure that screenshots are clear and readable and the renders should be at least 800px, but not more than 1600px at maximum.
- Sequential naming of images like, image 001.png... etc.
- Text should be in either ODT, DOC, TXT or HTML.
- Archive them using 7zip or RAR or less preferably zip.

3. Please include the following in your email:
- Name: This can be your full name or blenderartist avtar.
- Photograph: As PNG and maximum width of 256Px. (Only if submitting the article for the first time)
- About yourself: Max 25 words .
- Website: (optional)

Note: All the approved submissions can be placed in the final issue or subsequent issue if deemed fit. All submissions will be cropped/modified if necessary. For more details see the blenderart website.

Issue 28

"Up to Speed" in Blender 2.5 +

- Work flow
- Tips, techniques
- Tools: both new or improved tools
- Personal and or commercial projects in 2.5
- Surprise me!

Disclaimer